# Parsimonious Downgrading and Decision Trees Applied to the Inference Problem

**LiWu Chang** & **Ira S. Moskowitz**
Center for High Assurance Computer Systems, Mail Code 5540
Naval Research Laboratory
Washington, DC 20375
USA

ABSTRACT

In this paper we present our new paradigm for dealing with the inference problem which arises from downgrading. Our new paradigm has two main parts: the application of decision tree analysis to the inference problem, and the concept of parsimonious downgrading. We also include a new thermodynamically motivated way of dealing with the deduction of inference rules from partial data.

**Keywords**

Data mining, inference, downgrading, rules.

## 1. A NEW PARADIGM

Our new paradigm is a combination of *decision tree analysis* and *parsimonious downgrading*. Decision tree analysis has existed in the field of AI since the 1980's [3]. In brief, decision trees are graphs associated to data, with the goal of deducing rules from the data. Our new paradigm applies decision trees to the inference problem. In this paper we introduce the new concept of parsimonious downgrading. When High wishes to downgrade a set of data to Low, it may be necessary, because of inference channels, to trim the set. Parsimonious downgrading is a framework for formalizing this phenomenon. In parsimonious downgrading, we assign a cost measure to the potential downgraded information that is *not* sent to Low. We wish to see if the loss of functionality associated with not downgrading this data is worth the extra confidentiality. Decision trees assist us in analyzing the potential inference channels in the data that we wish to downgrade. We consider the confidence in rules produced by decision tree analysis. We analyze changes in confidence caused by missing data with a new theory we call the thermodynamic approach (which measures the changes in entropy). Our analysis is still at a preliminary stage and we wish to flesh it out with the participants of this workshop. In [6] rules are gleaned from rough set analysis of data, and the concept of not downgrading information, based upon inferences brought forth by these rules, is briefly introduced. We view [6] as motivation for some of our work on parsimonious downgrading. Since we prefer single-valued belief representations we do not use rough sets.

Our objectives in developing our new paradigm are:
**1**— Use decision trees (instead of rough set analysis) for the inference problem.
**2**— Make a study of not downgrading certain information.
**3**— Assign penalty functions to this parsimonious downgrading in order to minimize the amount of information that is not downgraded, and compare the penalty costs to the extra confidentiality that is obtained.
**4**—Take a thermodynamic approach to decreasing the confidence in rules that Low may infer from High data.

We believe that the current state of the art in the MLS community does not take advantage of statistical AI techniques. However, database researchers certainly do (there has also been some related work in intrusion detection). We want to change this by siphoning off valuable techniques from our sister sub-fields in computer science. Further, we feel that downgrading should be viewed as a flexible, rather than a static, process. We believe that our new paradigm is an attempt to change the status quo both in the use of statistical AI techniques (decision trees) and parsimonious downgrading.

### 1.1 Controversial?

We realize that the idea of changing the set of data that High wishes to downgrade might trouble some readers. If High has sanitized high-data into low-data, what is the problem? The problem is that the relations within this set of data might still be high. Of course, this has been noted in many papers. The paradigm that we wish to call into question is being "stuck" with the data that has been sanitized (and thus, is ready for downgrading). We hold that this data's value to Low must be weighed against the possible high-inferences that Low can deduce. If the information is of grave importance to Low, then it is downgraded. If some of it is of a lesser import and is outweighed by its loss of confidentiality, then perhaps some of the data can be trimmed from the set intended for downgrading. Downgrading should not be a static process — the trade-offs should always be measured. If functionality overrides confidentiality, then at least High is making an informed decision and is aware of the risk. Also, our techniques may be useful for machine-aided downgrading. Our concerns are not with whether Low and High are cooperating; rather, our concerns are with obtaining bounds for information leakage.

# Report Documentation Page

*Form Approved*
*OMB No. 0704-0188*



| 1. REPORT DATE | 2. REPORT TYPE | 3. DATES COVERED |
|---|---|---|
| **1998** | | **00-00-1998 to 00-00-1998** |

| 4. TITLE AND SUBTITLE | 5a. CONTRACT NUMBER |
|---|---|
| **Parsimonious Downgrading and Decision Trees Applied to the Inference Problem** | 5b. GRANT NUMBER |
| | 5c. PROGRAM ELEMENT NUMBER |
| 6. AUTHOR(S) | 5d. PROJECT NUMBER |
| | 5e. TASK NUMBER |
| | 5f. WORK UNIT NUMBER |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)<br>**Naval Research Laboratory,Center for High Assurance Computer Systems,4555 Overlook Avenue, SW,Washington,DC,20375** | 8. PERFORMING ORGANIZATION REPORT NUMBER |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) | 10. SPONSOR/MONITOR'S ACRONYM(S) |
| | 11. SPONSOR/MONITOR'S REPORT NUMBER(S) |

12. DISTRIBUTION/AVAILABILITY STATEMENT
**Approved for public release; distribution unlimited**

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | c. THIS PAGE | | | |
| **unclassified** | **unclassified** | **unclassified** | | **8** | |

## 2. A TOY EXAMPLE

For reasons of performance and functionality, it is sometimes necessary for High to downgrade information to Low. In this paper we will not get involved in the debate on whether this assumption is appropriate. (We accept it as a necessary evil, especially when dealing with databases [5].) Our example will be a database as given in Table 1. It is necessary for High to downgrade to Low the first eight rows in their entirety, and the ninth row with the result missing. If downgrading such a large amount of information bothers the reader, then just view the database as a small part of a larger set that High does not downgrade to Low. Note that we have modified an example from [15].

Table 1: High Database

| *Name* | *Hair* | *Height* | *Weight* | *Lotion* | *Result* |
|---|---|---|---|---|---|
| Hillary | blonde | average | light | no | burned |
| Janet | blonde | tall | average | yes | no |
| Bill | brown | short | average | yes | no |
| Tipper | blonde | short | average | no | burned |
| Newt | red | average | heavy | no | burned |
| Ken | brown | tall | heavy | no | no |
| Al | brown | average | heavy | no | no |
| Paula | blonde | short | light | yes | no |
| Tony | blonde | average | heavy | no | burned |

High has decided that the result for Tony should not be downgraded. Therefore, it sends Low the entire database with Tony's result left blank (we represent this with a question mark), see Table 2.

Low uses only the first eight rows of its database to form its rules. This is because these are the only complete rows (tuples). We refer to rows that are downgraded in their entirety as the *base set*, see Table 3.

Can High assume that the information it tried to keep hidden from Low is still hidden? If Low is stupid, this is true. However, say that Low analyzes the base set. Low will see that every blonde who did not use lotion got burned. Since Tony is a blonde who did not use lotion, Low now knows that Tony got burned. We formalize this rule as *(hair = blonde)*$\wedge$*(lotion=no)* $\Rightarrow$ *(result = burned)*, (read: IF *** THEN ***).

Why should we be concerned about some fancy AI way of deducing the rules? The reason is that this is a toy example. We could have cooked up a much more complicated example where the rule that we would need in order to determine the result would require an extensive search and correlation of the various database attributes. This falls into the area of datamining [6]. Unlike other knowledge-based work on the inference problem (e.g. [6,16]) we do not use a rough sets approach [9]. We propose using decision trees. Decision trees can handle large and noisy amounts of data and produce inference rules. It is not clear to us how effective rough sets are with respect to large and noisy data. Furthermore, when dealing with inconsistent data, rough sets give upper and lower approximations whereas decision trees give a probability. We feel more comfortable with probabilities because they are an effective representation of complex patterns of reasoning. (The purpose of this paper is not to contrast the two approaches; rather, it is to introduce decision tree analysis, in conjunction with parsimonious downgrading, as a new paradigm. We will return to decision trees in a later section.)

Table 2: Low Database = Downgrade

| *Name* | *Hair* | *Height* | *Weight* | *Lotion* | *Result* |
|---|---|---|---|---|---|
| Hillary | blonde | average | light | no | burned |
| Janet | blonde | tall | average | yes | no |
| Bill | brown | short | average | yes | no |
| Tipper | blonde | short | average | no | burned |
| Newt | red | average | heavy | no | burned |
| Ken | brown | tall | heavy | no | no |
| Al | brown | average | heavy | no | no |
| Paula | blonde | short | light | yes | no |
| Tony | blonde | average | heavy | no | ? |

Table 3: Base Set

| *Name* | *Hair* | *Height* | *Weight* | *Lotion* | *Result* |
|---|---|---|---|---|---|
| Hillary | blonde | average | light | no | burned |
| Janet | blonde | tall | average | yes | no |
| Bill | brown | short | average | yes | no |
| Tipper | blonde | short | average | no | burned |
| Newt | red | average | heavy | no | burned |
| Ken | brown | tall | heavy | no | no |
| Al | brown | average | heavy | no | no |
| Paula | blonde | short | light | yes | no |

The second part of our new paradigm is parsimonious downgrading. (Again, for the sake of integrity, we note that [6] contains a brief mention of this idea.) We see that in our example, Low will be able to deduce the rule *(hair = blonde)*$\wedge$*(lotion=no)* $\Rightarrow$ *(result = burned)* and thus determine that Tony gets burned. How can High prevent this? High can prevent this by not downgrading any information, but this is a bit of overkill. Instead, we feel that an approach that we call parsimonious downgrading should be used. In parsimonious downgrading, High decides what not to downgrade based upon the rules that it thinks Low can infer, and upon the importance of the information that Low should receive. If the information is of trivial value, it might also send incorrect data to Low (only for some attribute values) to impinge upon Low's ability to infer rules and therefore infer High information. High could decide not to downgrade both *Hillary-Lotion = no* and *Tipper-Lotion = no*. Then Low could not determine the rule *(hair = blonde)*$\wedge$*(lotion=no)* $\Rightarrow$ *(result = burned)* and the result concerning Tony would not be apparent to Low. What is the impact of not downgrading the information about Hillary's and Tipper's lotion? If, for functionality and performance reasons, Low must have this information, then there is a problem. If the importance of the information about Hillary's and Tipper's lotion is so great perhaps it is worth compromising the information about Tony's lotion use. This is worth thinking about. Security, as has been noted [4] need not be a yes/no world. Fuzziness might be appropriate in some cases. Perhaps it is extremely important for Low to know that Hillary did not use lotion but it is not really important for Low to know about Tipper's lotion use. Then High could downgrade everything as in Table 2 with the exception of *Tipper-Lotion*. This would result in what we call a *reduced downgrade*, as given in Table 4, for the Low database. How does this impact Low's rule making process?

Now we form the *reduced base set*, see Table 5. Unlike the original base set given in Table 3 we still include a row even though there is a unknown attribute value. This is because the result is still visible to Low. It is possible, though, that High, by parsimonious downgrading, decided to keep the result unknown to Low (not downgrade it to Low). Then we would not include that row in the reduced base set because it would not assist Low in forming a rule. Again, under parsimonious downgrading, deciding what to downgrade and what not to downgrade involves the functionality value of

Table 4: Low Database = Reduced Downgrade

| *Name* | *Hair* | *Height* | *Weight* | *Lotion* | *Result* |
|---|---|---|---|---|---|
| Hillary | blonde | average | light | no | burned |
| Janet | blonde | tall | average | yes | no |
| Bill | brown | short | average | yes | no |
| Tipper | blonde | short | average | ? | burned |
| Newt | red | average | heavy | no | burned |
| Ken | brown | tall | heavy | no | no |
| Al | brown | average | heavy | no | no |
| Paula | blonde | short | light | yes | no |
| Tony | blonde | average | heavy | no | ? |

Table 5: Reduced Base Set

| *Name* | *Hair* | *Height* | *Weight* | *Lotion* | *Result* |
|---|---|---|---|---|---|
| Hillary | blonde | average | light | no | burned |
| Janet | blonde | tall | average | yes | no |
| Bill | brown | short | average | yes | no |
| Tipper | blonde | short | average | ? | burned |
| Newt | red | average | heavy | no | burned |
| Ken | brown | tall | heavy | no | no |
| Al | brown | average | heavy | no | no |
| Paula | blonde | short | light | yes | no |

the information. Note that using the reduced base set in Table 5 and deleting the Tipper row, will still produce the same rules as before. However, our confidence in the rules concerning blondes has decreased, because the data backing our rule has decreased. The data, both in quality and quantity, should influence which rules are generated and the confidence we have in these rules. We note that we do not make the notion of confidence precise in this paper; however, it is part of our current research agenda. We will readdress this in the subsection on our thermodynamic approach.

## 3. DECISION TREE ANALYSIS

We continue with our toy example and attempt to formally determine the rules for what causes a sunburn. Consider the base set as given in table 3. What are the rules? (We have modified the example from [15] which is based on the work on ID3 [10,11].) This brings us to the more general discussion of what we mean by "the rules." From a given amount of data we need a way to generate inference rules. How can we be sure that no exception to the rule exists? We can't! The method we use for generating rules is statistical in nature. In fact, we will show two possible decision trees (which we use to read off the rules) for the same data. We use an information theoretical approach [10] to generate our decision trees. We believe this is be a realistic approach. Note that we are presently working on allowing this information theoretical approach to incorporate Bayesian techniques. We feel that this will allow us to adjust our given data against our preconceived notions of the appropriate prior probabilities. Here, we will not use Bayesian techniques for reasons of (1)—simplicity, and (2)—we have yet to formalize the application.

Shannon first put information theory on a firm foundation [12]. We use his concepts of entropy and mutual information. The columns Hair, Height, Weight, and Lotion make up the attributes. We wish to see which has the greatest influence upon the result. To determine this we use the conditional entropy. Let $A$ be the random variable representing an attribute (we have four choices for this random variable) which takes on the values $a_i$ and let $R$ be the random variable representing the result which takes on the values $r_1$ = burned, and $r_2$ = no burned. We need to determine the mutual information $I(R, A)$ between the result and the attribute (use base two for the logs):

$$I(R, A) = H(R) - H(R|A)$$

where

$$H(R) = -\sum_j p(r_j) \log p(r_j)$$

and,

$$H(R|A) = -\sum_i p(a_i) \sum_j p(r_j|a_i) \log p(r_j|a_i)$$

The probabilities are determined by a frequency count based on the data. The attribute that has the most effect upon the result is the attribute that has the greatest mutual information. Since $H(R)$ is constant and $H(R) \geq H(R|A)$, the optimization condition is equivalent to finding the attribute that minimizes the conditional entropy $H(R|A)$. Thus we have the following:
**Gain Condition**[Quinlan]: Find $A$ such that $H(R|A)$ is minimized.

Let us take the first attribute $A$ = Hair, $a_1$ = blonde, $a_2$ = brown, and $a_3$ = red. This gives us
$H(R|A) = -\frac{4}{8}\left[\frac{2}{4}\log\frac{2}{4} + \frac{2}{4}\log\frac{2}{4}\right] - \frac{3}{8}\left[\frac{0}{3}\log\frac{0}{3} + \frac{3}{3}\log\frac{3}{3}\right] - \frac{1}{8}\left[\frac{1}{1}\log\frac{1}{1} + \frac{0}{1}\log\frac{0}{1}\right] = .5$
Similarly, we see that $H(R|\text{Height}) = .69$, $H(R|\text{Weight}) = .94$, and $H(R|\text{Lotion}) = .61$. Thus we see that the attribute that has the most influence upon Result is Hair.

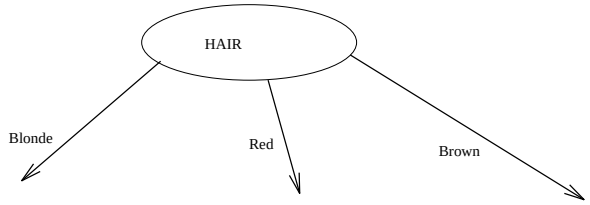


Figure 1. The First Branching

Now we must repeat the process for each node, until there are no more decisions to be made. Since every Red is Burn, and every Brown is no, those decisions are done. However, blonde is still not decided upon so we must find another attribute that "maximally" influences result.

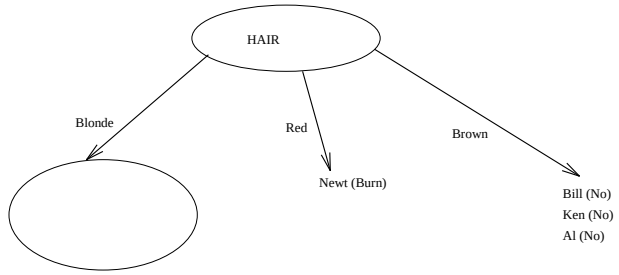


Figure 2. The First Branching with Partial Decisions

Now we must repeat the gain condition but we restrict ourselves to the blondes. So we must minimize $H(R|A)$, where $A$ = Height, Weight, or Lotion. Let us try Lotion, $a_1$ = no, and $a_2$ = yes. So, $H(R|\text{Lotion, Hair} = \text{Blonde}) = -\frac{2}{4}\left[\frac{2}{2}\log\frac{2}{2} + \frac{0}{2}\log\frac{0}{2}\right] - \frac{2}{4}\left[\frac{0}{2}\log\frac{0}{2} + \frac{2}{2}\log\frac{2}{2}\right] = 0$. We need

not calculate any other conditional entropies. All they can do is tie (but they do not). So the next attribute we put down as a node is Lotion.

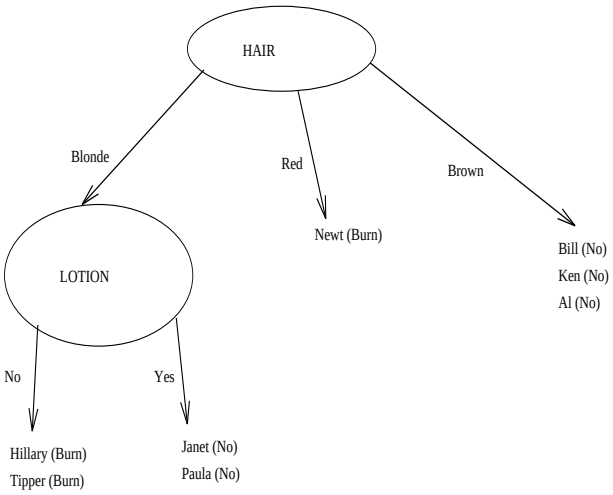



Figure 3. Decision Tree

Now we can read off the following four rules:
*(hair = blonde)∧(lotion=no) ⇒ (result = burned)*
*(hair = blonde)∧(lotion=yes) ⇒ (result = no)*
*(hair = brown) ⇒ (result = no)*
*(hair = red) ⇒ (result = burn).*
We see that the first rule is the obvious one that we discussed before. At this stage we can actually reduce the rules down to smaller set. There are several ways of accomplishing this. One uses Fisher's exact test [15] to determine more finely the sensitivity of the various attributes. Another approach is to prune and rebuild the tree [11], where the pruning is accomplished by analyzing the predictive power of the original tree. It is not the purpose of this paper to go into this in detail. Rather, it is our intention to show the new paradigm of decision trees applied to parsimonious downgrading.

Note if we did not use the gain condition, but simply built a decision tree based on logical inferences we could end up with a tree [15] that gives "strange" rules.[1]

The rules that we produce must be tempered with a confidence level. We do not go into details here but consider the rule *(hair = red) ⇒ (result = burn)*. This is based on a single tuple. How much confidence can we place in this rule? The more data that supports a rule, the more confidence we have in its predictive powers.

What if we did not use the gain condition to propagate our rules? Consider Figure 3.5, a perfectly valid tree of inferences. However, how useful are the rules that Low could derive from it? Consider the rule
*(height=tall)∧(weight=heavy)∧(hair=red) ⇒(result = no).*
This is a valid but pretty useless rule, because it has too many antecedents. The gain condition minimizes the number of antecedents because of the minimal entropy condition. We also see that our knowledge about sunburns is not expressed in Figure 3.5. We know that hair color and the use of lotion (SPF $\geq$ 15) affects hair color. This is why we are advocates of Bayesian estimation. We are allowed to express this knowledge by the use of priors, along with the data on hand.

[1] Again, we feel that Bayesian techniques should be used in conjunction with the gain condition so our prior belief in certain conditions can influence the rule-making process.

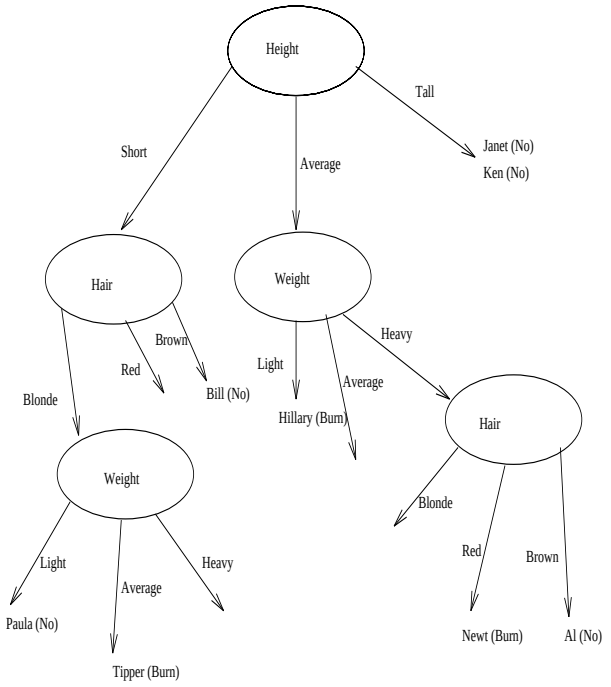



Figure 3.5.

Now that we have what we believe to be a statistically valid way of determining rules, we wish to use these rules to see how downgrading is affected.

As time goes on, further data may be downgraded from High to Low. Our rules can be refined to take this new data into account. Decision trees can be regenerated from the new data. However, this might be computationally unfeasible. (Should we keep a record of all of this old data?) In that case, we could use our new data, and statistical updating procedures [1], to refine the confidence that we have in our rules.

### 3.1 Recap

We use decision rules because they have proven to be fruitful and accurate predictors in the AI world [11, 15]. They are computationally feasible and they have a firm information theoretical foundation. If Low uses other methods, Low can certainly produce rules but we feel that these rules will not be stronger predictors than the rules produced via decision trees. Therefore, security arguments based on decision trees will be conservative. To further strengthen our analysis, we are at present comparing how our rules generalize to other methods.

## 4. PARSIMONIOUS DOWNGRADING

Our concern is the following: High and Low exist in separate worlds.

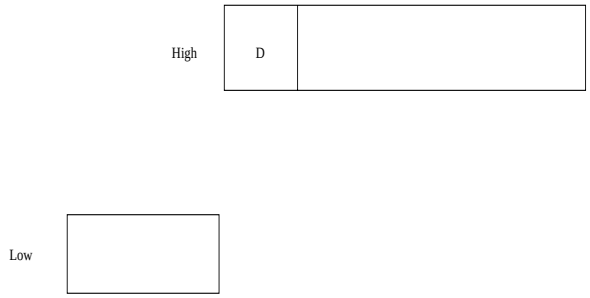



Figure 4.

High wishes to downgrade the set $D$ to Low for reason of system functionality.

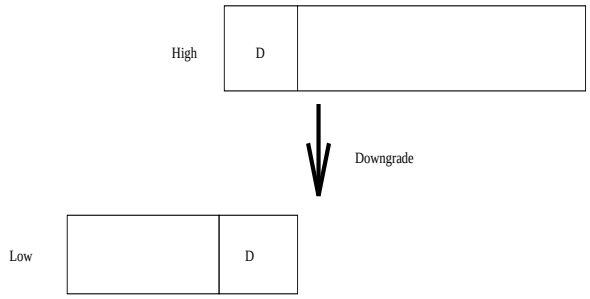



Figure 5.

Low, by using decision tree analysis (or, if preferred, some other method, e.g. Pawlak's rough set approach [6,16]), is able to determine rules that will enable Low to infer high-data outside of the set $D$.

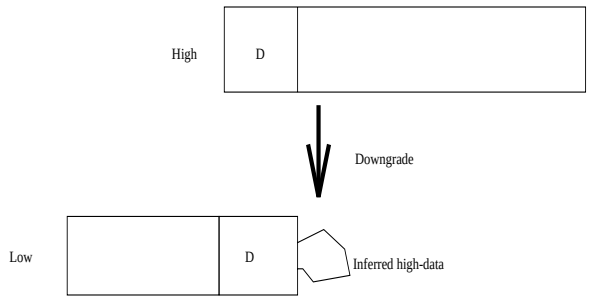



Figure 6.

High also knows what rules Low can determine and decides not to downgrade $D$ but rather $D' \subset D$, $D' = D - d$ . Pictorially, we view $D'$ as the set $D$ with the black spot (which is $d$) in it.

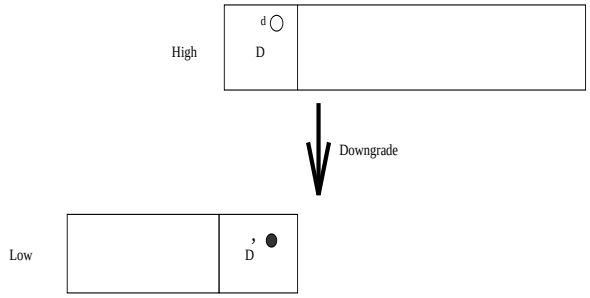



Figure 7.

We must determine the trade-offs between Low not receiving $d$ and the insecurity caused by Low obtaining the inferred high-data.

What is the importance to functionality of Low obtaining $d$? Is it enough to cause system failure, or is it something that just slows down systems performance? Is the set $d$ of a milk/wine-nature (it is important now but not in the future/it is important in the future but not now). Does the importance of $d$ oscillate throughout time? Which set $d$ do we choose? How should one measure the impact of $d$ upon the inferred high-data? What are the security concerns if Low receives inferred high-data? Are they extremely grave or are they just a minor security leak? Is the threat constant throughout time? — Or is the threat dynamic in nature? These ideas are a starting point for this part of our new paradigm.

The elements of $D$ should not be viewed in isolation for either their functionality purposes or their security purposes. We see that in databases an attribute value alone is not as important as a tuple of attribute values. Also, we have discussed the dynamic nature of both system functionality and insecurity/security. At this meeting last year, the notion of insecurity flow and the effects of time upon insecurity were noted [7]. Let $\iota$ denote the insecurity that may occur. Let $F$ denote the system functionality of Low. We realize that these concepts are not well-defined. However, we feel that they are sufficiently well-defined to continue with our trade-off discussion. Since the elements of $D$ should not be considered in isolation and time is affecting both functionality and insecurity, we define the following two functions (possibly relations?):

$$L : 2^D \times T \times M \rightarrow \iota$$

where $2^D$ is the power set of $D$, $T$ is time, and

$$U : 2^D \times T \times M \rightarrow F$$

$U$ is acting as a utility function and $L$ is representing security leaks. The set $M$ takes into account factors we are not aware of—this could be system load, changes in computers composing a distributed system, extra security measures that vary in time, etc. It is possible that the factors constituting $M$ are actually taken into account via $T$ but we wished to include $M$ to give us some wiggle room for unknown factors. We assume that both $\iota$ and $F$ are have some sort of measure (such as the volume of a set) or metric (such as the magnitude of an element) (distinct for each set) on them so we can judge what has more insecurity or functionality.[2] As an example, $\iota$ could be the node insecurity as in [7].

$L$ and $U$ should both have the properties of being non-decreasing with respect to inclusion on their domain sets, e.g., if $A \subset B \in 2^D$ then $L(A) \leq L(B)$ and $U(A) \leq U(B)$.[3]

We wish to make trade-offs between $L$ and $U$. Specifically, we wish to compare to the images $L(D) \subset \iota$ and $U(D) \subset F$ with those of $L(D')$ and $U(D')$. Our goal is to determine if the insecurity difference between $L(D)$ and $L(D')$ is worth the loss of functionality between $U(D)$ and $U(D')$. How do we measure the differences between $U(D)$ and $U(D')$? In our toy example, perhaps the Lotion use is extremely important to Low's functionality, but the Weight

[2] We realize that this is controversial. We would like to discuss this at the workshop and refine these sets in future versions of this paper/work.

[3] When we write $L$ or $U$ as a function of just the first variable, it is understood that the values for $T$ and $M$ are fixed and are not germane to the discussion at this point.

is much less important to Low's functionality. Also, as we have discussed and shown in our notation, these differences may vary in time. How do we measure the added insecurity obtained by downgrading $D'$ instead of $D$? In our toy example, we know that Tony is a blond but do not know the result. Therefore, when we form the set $d$ it should be made up of blondes. However, this type of problem assumes that we are concerned about Low inferring information at the present time from the set of data that High has downgraded. What if our concern is the very fact that Low can propagate rules from downgraded information and possibly use those rules in the future to infer data? We draw this distinction because in our toy example we are concerned with a blonde, but in this new way of thinking, perhaps High in the future will downgrade some partial information about a red head, and we do not want Low to infer the result about *that* future red head. Therefore we wish to make the rules opaque that Low may infer, and thus mislead Low. This should be reflected in the mapping given by $L$. One possible way to do this is to again invoke entropy and maximize the amount of confusion. This leads us to the next section.

## 5. A THERMODYNAMIC APPROACH

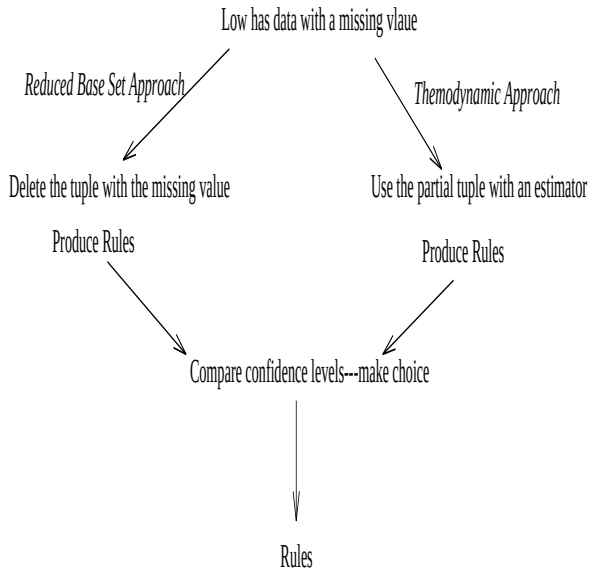
Our concern (as noted above) is to mitigate the confidence in the various rules that Low can infer. Our concern is with the predictive powers of the rules in general without regard to any specific question that Low may attempt to answer. In this subsection, we present our own (not completely formalized) theory and invite feedback from the workshop participants. Given a set $D'$, as before, what is the best that Low can do with this set? We present our new approach, the thermodynamic approach, as a way for Low to deduce rules from the diminished data with high levels of confidence. In other words, we feel that our approach maximizes the leakage function $L$.

In our method, one forms the decision tree as in section 3, by using the gain condition, and minimizing the conditional entropy at each stage. When finished, one will have a decision tree with only certain attributes as nodes. Call these attributes $A_1, \cdots, A_n$. Consider the entropy ($R$ is still the result you are interested in) $H(R|A_1, \cdots, A_n)$. The value of the term $H(R|A_1, \cdots, A_n)$ is called the *initial temperature*, $\tau_0$, of the data (At this point we are still investigating which attributes to condition on. For the sake of simplicity we condition on all $n$ in this section. However, this analysis requires further work and this section should be viewed as work in progress.). We wish to perturb the data in order to raise the temperature. This will lower the confidence of the various rules that are generated from the decision tree. Our perturbation is not done by introducing erroneous data (this could be done and we will explore this in future work). Instead, the perturbation is done by deleting data so that Low (as before) is missing data. Of course, this deletion of data must be done in a value added way keeping the utility function $U$ in mind. The method propagates a probabilistic decision tree by Low using parameters for the missing data. We calculate the new value of $H(R|A_1, \cdots, A_n)$ and call it the *present temperature*, $\tau_p$. We are interested in $\Delta\tau = \tau_p - \tau_0$.

Our approach is motivated by the thinking behind behind Quinlan's gain condition and the third law of thermodynamics [14] ($\approx$ as thermal motion decreases, so does entropy decrease).

Before, when we discussed the reduced base set in Table 5, we said that by deleting the Tipper tuple, we could still form a decision tree and produce rules. The rules would be the same as what we originally had but the confidence in the "blonde" rules would be lessened. We do not put a metric on the rules (work like this has been done in [11]). However, we do point out that this (undescribed) decrease in confidence must be compared to $\Delta\tau$. This will give Low a probabilistic way of dealing with the missing data and producing the "best" rule set possible under those conditions. This will also give High guidance in how to delete data from the set to be downgraded.

Ideally, High does not want to downgrade large amounts of data. With this in mind, if High then performs parsimonious downgrading and sends both small and noisy data down to Low, Low would want to take advantage of as much data as possible. Therefore, Low would not want to delete tuples with missing data but would instead use an approach, such as our thermodynamic approach, to use the already sparse data that it has.



Low deciding which approach to use

Note that both branches use decision trees! Confidence levels for the left branch can be taken from standard statistical non-parametric methods [2]. However, we do not have a theory for the right hand branch and are looking at the problem. Also, we assume that High is as good a statistician, information theorist, AI engineer, etc., as Low. But High has the dual job of attempting to mitigate Low's rule producing and trying to give Low as much functionality as possible.

EXAMPLE: For the sake of brevity and clarity, we define our method by example. Consider that Table 5 has the reduced base set. Now, instead of deleting the Tipper row we will use it by putting a parameter into the Lotion column. We call the parameter $\theta$, $0 \leq \theta \leq 1$. The parameter represents a probability for one of the possible attribute values. We are assuming that it represents No Lotion. However, it is really a second order probability. By this we mean that $\theta$ itself is given by a distribution. This is done so Low can attempt to use as much given information as possible. Now we have a *parametric base set*.

As stated above, the $\theta$ in the Lotion column is to be read as $P(No = \theta)$ and $P(Yes = 1 - \theta)$. As before, we must

Table 6: Parametric Base Set

| *Name* | *Hair* | *Height* | *Weight* | *Lotion* | *Result* |
|---|---|---|---|---|---|
| Hillary | blonde | average | light | no | burned |
| Janet | blonde | tall | average | yes | no |
| Bill | brown | short | average | yes | no |
| Tipper | blonde | short | average | $\theta$ | burned |
| Newt | red | average | heavy | no | burned |
| Ken | brown | tall | heavy | no | no |
| Al | brown | average | heavy | no | no |
| Paula | blonde | short | light | yes | no |

apply the gain condition and minimize $H(R|A)$, where $A$ is Lotion, Weight, or Height. Let us calculate $H(R|Lotion)$. There are $4+\theta$ no's in the Lotion column and $3+(1-\theta)$ yes's. The probability of a No Lotion not being burned is $\frac{2+\theta}{4+\theta}$; the other probabilities follow similarly. Thus we have that

$$H(R|Lotion) = -\frac{4+\theta}{8}\left(\frac{2+\theta}{4+\theta}\log\frac{2+\theta}{4+\theta} + \frac{2}{4+\theta}\log\frac{2}{4+\theta}\right) - \frac{4-\theta}{8}\left(\frac{1-\theta}{4-\theta}\log\frac{1-\theta}{4-\theta} + \frac{3}{4-\theta}\log\frac{3}{4-\theta}\right)$$

The minimum of this function occurs when $\theta = 1$, and it is .61. Since .61 is still greater than .5 $= H(R|Hair)$, the first node is still Hair. What about the second node? For this we have that $H(R|$Lotion, Hair = Blonde$)$ reduces to $-\frac{2+\theta}{4}\left(\frac{\theta}{2+\theta}\log\frac{\theta}{2+\theta} + \frac{2}{2+\theta}\log\frac{2}{2+\theta}\right)$. This function ranges from 0 to about .7. Unfortunately, it is not always less than $H(R|$Height, Hair = Blonde$) = .5$. It depends on the value of $\theta$. For $\theta$ values such that it is less than .5, Lotion would be our secondary node. For the other values it would be Height, and we see that we get a different rule set. With either decision tree, we calculate the new temperature for the rules and weight the rules by $\Delta\tau$. We do not have the weighting figured out yet. However, we feel that this is the correct approach based upon the known statistical results.

We have not discussed how to pick values for $\theta$. One way is by using Bayesian estimators (e.g., [1]). This is a valid statistical method that incorporates the given data along with some prior belief in the probabilities to avoid over-fitting of the data. The Bayesian approach lets one assign probabilities in a manner that minimizes the risk of error. For example, for our missing Lotion use of Tipper, assuming a non-informative (uniform) prior we would have a probability of 2/5 for no Lotion use. Bayesian techniques can also be used to give a realistic range of probabilities, and thus give further guidance towards the confidance levels associated to Low's rules.

One might argue that the very fact that High is hiding information from Low can, in fact, be sending information to Low. This seems to be more of a psychological than statistical attack. We invite comments from the workshop participants upon this. Note that preconceived notions can be accounted for in the assignment of the prior distribution.

## 6. TRADE-OFFS

After parsimonious downgrading has been performed, Low can produce rules and those rules have a confidence level associated with them. That confidence level goes into the calculation of $L$, the leakage formula. On the other hand parsimonious downgrading affects $U$ the utility function of the data that is downgraded. We must see if the increased security is worth, in High's mind, the functionality hit that Low will take.

In essence, we have a dynamic programming constraint-based problem. The loss of security (increases in $\iota$) must be balanced against the decreases in functionality. In Figure 8 we see the image of the function $U \times L$, where

$$U \times L : 2^D \times T \times M \times 2^D \times T \times M \rightarrow F \times \iota\ .$$

We are interested in the pre-image from the lower right hand region (the feasible region) of $F \times \iota$ space. For $D'$ to meet both the minimum functionality and minimum security requirements it is necessary that

$$D' \in \Pi_1\left((U \times L)^{-1}\,(feasible\ \ region)\right) \cap\ \Pi_4\left((U \times L)^{-1}\,(feasible\ \ region)\right)\ ,$$

where $\Pi_i$ is projection into the $i$th factor.[4] Keep in mind that $D'$ produced in this manner is a refinement over our original concept of $D'$. At the start of this paper we were just concerned with diminishing the set $D$ in order to lessen Low's inferencing capabilities. Now we also want to include functionality requirements in our production of $D'$. It is possible for the image of the function $U \times L$ to not intersect the feasible region. In this case we would not have any candidate for $D'$ that met both our security and functionality requirements. The $D'$ produced in this manner are the ones that we attempt to balance security against functionality. $D'$ not produced in the above manner have either (or both) intolerable insecurity or intolerable lack of functionality.

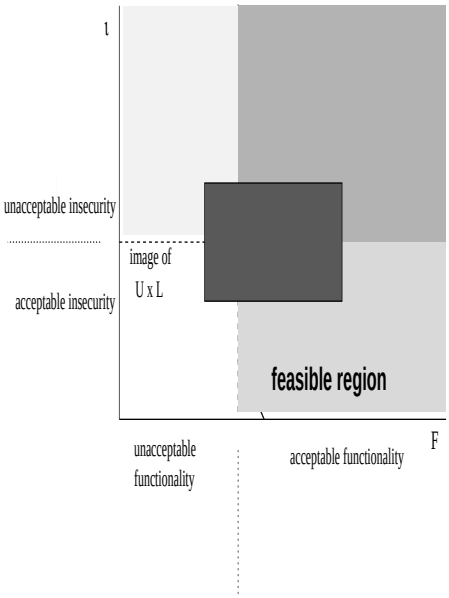



Figure 8. Heuristic Representation of Feasibility Region

Considering the temporal nature of the downgrading, a stochastic game theoretic [13] approach might be called for. Consider a two-person game where the gains are the increase in security $(-L)$ and the losses $(-U)$ are the decrease in functionality. From this, one should be able to produce a pay-off function. We feel that it will be a very complicated game and we will most likely not have a zero-sum game because the gains might not equal the losses.

[4] As in Footnote 3 we are implicitly assuming (for simplicity) that $T$ and $M$ are fixed at $(t, m)$. The inverse image and projection of feasible regions should really be done for each choice of $(t, m)$.

As noted, there are other methods for generating rules aside from decision trees. It is possible that Low can use a smorgasbord of techniques. Then how High performs its parsimonious downgrading must be reviewed in this light. We are confident that decision trees give a conservative view (which is what we want). However, we want to compare non-decision tree based techniques also. We plan to study this issue to see if our decision tree approach is really as strong as we feel it is, or if other methods must be considered in conjunction with decision trees when considering leakage versus utility of downgraded data.

Other ideas are welcomed from the workshop participants.

## 7. SUMMARY

We have presented our new paradigm, which consists of several parts. Some of these parts are well-grounded in other areas (decision tree analysis), but have never been applied by our community. Some are totally new ideas (parsimonious downgrading) but it is not clear how to formalize the associated utility and leakage functions. The thermodynamic approach to dealing with base sets after High has deleted data is the most controversial part of this paper. We believe in it, but have not yet proved it. Discussion with the participants will help us to refine the details, or cause us to go another way. Either way we feel that this paper is a new approach to dealing with the inference problems caused by downgrading. We also feel that our new paradigm will be useful in the more general (and recently very active) field of datamining in general.

In future work we want to investigate OR techniques and the use of utility functions in analyzing trade-offs. Also, when analyzing trade-offs we wish to study how measurements of bits of correct vs. incorrect data, and standard correlation analysis may come into play. Instead of just deleting data we might want to corrupt some of the data, but this comes at the cost of integrity and must also be studied. Also changes in strategies can be taken into account by varying the Bayesian parameters.

Note that this paper is part of a project, which we have recently started, on Knowledge Discovery and Datamining (KDD) applied to secure systems. Our other papers of interest are, as of this date, [1] which we have already mentioned, and [8] which takes an approach similar to perfect secrecy in order to analyze the database inference problem.

## 8. ACKNOWLEDGEMENTS

We thank Judy Froscher, Myong Kang, Cathy Meadows, and Bruce Montrose for their helpful discussions. We are especially grateful to Ruth Heilizer for her careful reading of the draft versions of this paper. We also thank the anonymous referees for their helpful comments. Research supported by the Office of Naval Research.

## 9. REFERENCES

[1] Chang, L., and Moskowitz, I.S. (1998) "Bayesian Methods Applied to the Database Inference Problem," *Proc. IFIP WG11.3 Working Conference on Database Security*, Greece, 1998.

[2] Dudewicz, E.J., and Moshra, S.N. (1988) *Modern Mathematical Statistics*, Wiley.

[3] Feigenbaum, McCorduck, & Nii (1988) *Rise of the Expert Company*, Times Books.

[4] Hosmer, H. (1993) "Security is Fuzzy!," *Proc. New Security Paradigms Workshop*, pp. 175-184, 1993.

[5] Kang, M.H., Moore, A. & Moskowitz, I.S. (1998) "Design and Assurance Strategy for the NRL Pump," *Computer*, IEEE CS Press, April, pp. 56-64, 1998.

[6] Lin, T.Y., Hinke, T.H., Marks, D.G., & Thuraisingham, B. (1996) "Security and Data Mining," *Database Security Vol. 9: Status and Prospects,* IFIP, pp. 391-399.

[7] Moskowitz, I.S. and Kang, M.H. (1997) "An Insecurity Flow Model," *Proc. New Security Paradigms*, pp. 61-74, 1997.

[8] Moskowitz, I.S. and Chang, L. (1999) "A Formal view of the Database Inference Problem," *Proc. CIMCA'99*, 1999.

[9] Pawlak, Z. (1982) "Rough Sets," *Int. J. Comput. Inf. Sci.*, 11. pp 341-356, 1982.

[10] Quinlan, J.R. (1979) "Discovering Rules by Induction from Large Numbers of examples: A Case Study," *Expert Systems in the Micro-Electronic Age*, D. Michie, editor, Edinburg Univ. Press.

[11] Quinlan, J.R. (1993) *C4.5 Programs for Machine Learning.* Morgan-Kaufmann.

[12] Shannon, C. (1948) "A Mathematical Theory of Communication," *Bell System Technical Journal*, Vol. 27, pp. 379-423, 623-656, July, October, 1948.

[13] Syverson, P.F. (1997) "A Different Look at Secure Distributed Computation," *Proc. Computer Security Foundations Workshop*, pp. 109-115, 1997.

[14] Waldram, J.R. (1985) *The Theory of Thermodynamics*, Cambridge University Press.

[15] Winston, H. (1991) *Artificial Intelligence*, 3rd Ed., Addison-Wesly.

[16] Zhang, K (1997) "IRI: A Quantitative Approach to Inference Analysis in Relational Databases," *Proc. IFIP WG 11.3 Working Conference on Database Security, 1997.*